International Journal of Informatics and Communication Technology (IJ-ICT)
Vol. 12, No. 2, August 2023, pp. 127~139
ISSN: 2252-8776, DOI: 10.11591/ijict.v12i2.pp127-139

Machine learning techniques for plant disease detection: an evaluation with a customized dataset

Amatullah Fatwimah Humairaa Mahomodally, Geerish Suddul, Sandhya Armoogum
Department of Business Informatics and Software Engineering, School of Innovative Technologies and Engineering, University of Technology, Port Louis, Mauritius

Article Info

Article history:

Received Oct 4, 2022
Revised Nov 28, 2022
Accepted Dec 30, 2022

Keywords:

Convolutional neural networks
Deep learning
Plant disease detection
Transfer learning

ABSTRACT

Diseases in edible and industrial plants remains a major concern, affecting producers and consumers. The problem is further exacerbated as there are different species of plants with a wide variety of diseases that reduce the effectiveness of certain pesticides while increasing our risk of illness. A timely, accurate and automated detection of diseases can be beneficial. Our work focuses on evaluating deep learning (DL) approaches using transfer learning to automatically detect diseases in plants. To enhance the capabilities of our approach, we compiled a novel image dataset containing 87,570 records encompassing 32 different plants and 74 types of diseases. The dataset consists of leaf images from both laboratory setups and cultivation fields, making it more representative. To the best of our knowledge, no such datasets have been used for DL models. Four pre-trained computer vision models, namely VGG-16, VGG-19, ResNet-50, and ResNet-101 were evaluated on our dataset. Our experiments demonstrate that both VGG-16 and VGG-19 models proved more efficient, yielding an accuracy of approximately 86% and a f1-score of 87%, as compared to ResNet-50 and ResNet-101. ResNet-50 attains an accuracy and a f1-score of 46.9% and 45.6%, respectively, while ResNet-101 reaches an accuracy of 40.7% and a f1-score of 26.9%.

Corresponding Author:

Amatullah Fatwimah Humairaa Mahomodally
Department of Business Informatics and Software Engineering
School of Innovative Technologies and Engineering, University of Technology
Ave De La Concorde, Port Louis, Mauritius
Email: amahomodally@umail.utm.ac.mu

1. INTRODUCTION

Plant diseases affects global food production, biodiversity, our health and the livelihoods of farmers [1]–[7]. The correct usage of pesticides to preserve yields requires a high level of expertise since the indication of a particular disease varies from one plant species to another [2]. Even experienced plant pathologists may fall short in diagnosing diseases correctly, resulting in chemicals such as bactericides, fungicides and nematicides being used excessively, thus adversely affecting biodiversity [8]–[13]. Their use is also harmful to our health, causing both acute and chronic consequences such as neurological and metabolic deficits [2]–[5], [13]. Furthermore, the United Nations estimated that toxic exposure causes an average of 200,000 deaths per year [14]. On the other hand, in developing countries, the livelihoods of smallholder farmers who generate more than 80% of the cultivation are disastrously constrained by yield loss, which is reported to be more than 50% per year due to pests and diseases [15], [16].

Most of the time, diseases in plants are first detected by experienced farmers when they become visible. Though trained raters can detect the ailments with their naked eyes, their analysis may be erroneous

as they can be subjected to fatigue or loss of concentration since the necessary process of continuous plant monitoring is tedious and time-consuming, and harvests may expand over vast areas [17]. Plant disease detection can also be conducted through techniques such as enzyme-linked immunosorbent assay (ELISA), deoxyribonucleic or acid ribonucleic acid (DNA or RNA) probes, squash blots, tissue blotting and polymerase chain reaction (PCR) by distinguishing DNA or proteins that are different for each disease [18]–[21]. However, though the molecular test kits can detect diseases promptly, their development is expensive and they can be inaccessible to smallholder farmers [18]–[21]. On the other hand, to identify plant diseases quickly, researchers at North Carolina State University developed a sensory device to sample the airborne levels of volatile organic compounds (VOCs) that plants' leaves release [21]. Sensor-based methods have also been adopted to identify plant diseases by detecting early changes in plant physiology such as changes in leaf shape, tissue color and transpiration rate [20], [22], [23]. Nevertheless, accessibility, cost-benefit and training time are some factors that negatively affect the successful implementation of these technologies by smallholder farmers [21]. Furthermore, due to their efficiency in computer vision, various works in recent years have proposed the use of deep learning (DL) algorithms to detect diseases in plants. In the context of plant classification, DL performs classification more accurately than traditional machine learning (ML) methods [24]–[27]. However, further DL research in plant diseases is required in order to produce functional systems that can be utilized in practice [28]. The aim of this paper is to evaluate the state-of-the-art ML models to identify plant diseases and contribute to alleviating drawbacks encompassing plant disease detection, which could be of help to researchers in the field of computer vision for plant disease detection.

2. LITERATURE REVIEW

Application of ML methods in the agricultural sector to detect plant diseases has shown tremendous success [29]–[32]. According to Zhang *et al*. [33] ResNet performed better than GoogleNet and AlexNet for detecting tomato plant disease. Research by Türkoğlu and Hanbay [34] ResNet-50 with support vector machine (SVM) classifier performed efficiently in terms of f1-score and recall when classifying eight different plant diseases. Research by Ferentinos [35] deployed convolutional neural network (CNN) models such as AlexNet, AlexNetOWTbn, GoogLeNet, OverFeat, and visual geometry group (VGG), by using 87,848 images, VGG reached a relatively higher accuracy of 99.53% when classifying plant diseases.

Research by Jiang *et al*. [36] the VGG-inception architecture outranked ResNet, AlexNet, GoogLeNet and VGG in performance when classifying five types of apple plant diseases. Research by Nachtigall *et al*. [37] AlexNet performs better than multi-layer perceptron (MLP) with an accuracy of 97.3% when classifying diseases in apple plants from a dataset of 1,450 records. Research by Türkoğlu and Hanbay [34] AlexNet outperformed SqueezeNet in terms of accuracy when distinguishing tomato plant disease. Research by Brahimi *et al*. [24] AlexNet and GoogleNet perform better than classification techniques such as SVM and random forest, and AlexNet model reaches a relatively high accuracy of 99.18% for tomato disease detection.

Research by Kawasaki *et al*. [27] a CNN architecture that utilizes the Caffe framework [38] was proposed to detect diseases in the leaves of cucumbers. 800 images of leaves were used as dataset which was then augmented through rotational transformations. The study attained an accuracy of 94.9%. Research by Jia *et al*. [39] image rotations, and perspective transformations were used to enlarge 4,483 original images from the Stanford background dataset [40], after which transfer learning using CaffeNet architecture [38] was implemented to classify 13 types of diseases in plants. According to Rangarajan *et al*. [41] AlexNet has better accuracy compared to VGG-16 when determining six types of diseases in tomato plants [42]. Research by Mohanty *et al*. [6] when using the PlantVillage dataset with 38 classes, transfer learning of GoogleNet achieved an accuracy of 99.35%.

Overall, image recognition is extremely useful due to its ability to handle a large number of input parameters, such as image pixels [29]. It has a fast processing time, and it also lessens human efforts and errors [43]. In due course, computer vision through ML can be effectively used by farmers or even inexperienced users [29].

Nevertheless, various limitations were encountered in several studies. In many cases, the datasets used consist solely of images taken in laboratory-controlled environments and not in real-world setups [28], [35]. An evaluation of trained models utilizing plant images that include real-world environments showed a significant reduction in accuracy by 31% [28]. Another major setback of CNN for identifying specific diseases in plants is that existing public datasets consist of limited records and classes, and as such, cannot identify the vast variety of plants' ailments [29], [44]. Experimental results indicate that the use of datasets with small records prevents neural networks from properly learning the classes [29], [45], [46].

3. MATERIALS AND METHOD

This section covers the method used to implement the novel customized dataset and apply DL models for plant disease detection. The experiments were conducted with Jupyter notebook by using Keras with Tensorflow as backend on a simple model laptop without graphics processing unit (GPU). Figure 1 shows a high-level view of the architecture of our proposed solution.

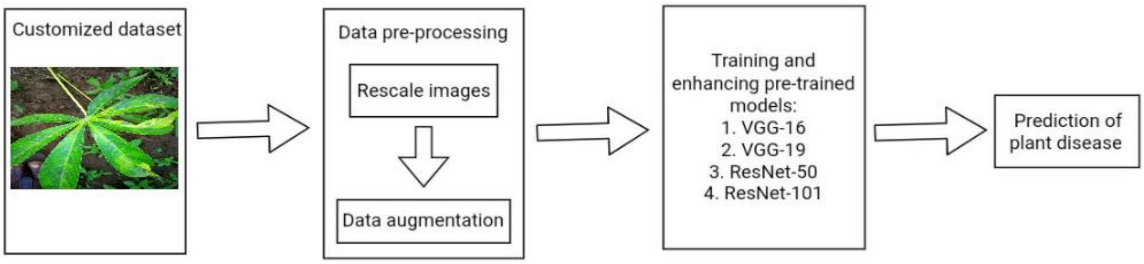

Figure 1. High level architecture of the proposed solution

3.1. Dataset

A customized dataset of 87,570 leaf images, under both lab-controlled and real-world conditions, across 32 crop species segregated into 97 distinct classes of healthy and diseased plants, was compiled for this paper as shown in Figure 2 and Table 1 (see in Appendix) by cleaning and combining multiple open datasets together. Among the 97 categories, 74 and 23 classes belong to diseased and healthy plants, respectively. The main issue with many existing open source datasets is that they consist of images of leaves assessed under lab-controlled setups only and often contain a small number of records and classes that are not appropriate for real-world applications. Our customized dataset consists of a rather large set of records, which can definitely help to overcome the issue of overfitting. Overfitting is a major problem linked to small datasets that consequently produce less reliable models that do not generalize well. Moreover, our customized dataset results in an interesting mix of both lab-controlled and real-world images due to the various open datasets that we used, as depicted in Table 2.

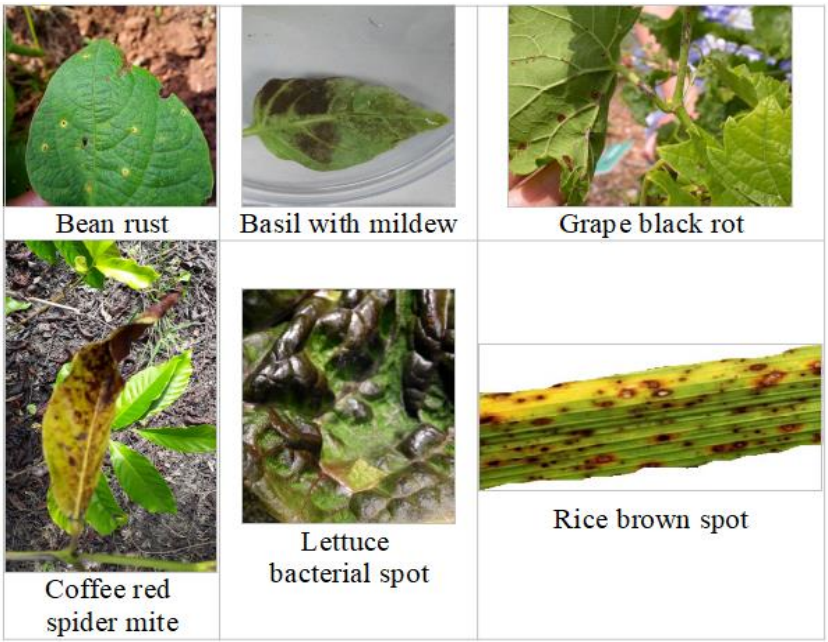

Figure 2. Some leaf images from our customized dataset

Table 2. Description of datasets from which leaf images were adopted

Dataset name	Dataset description	Environment	Number of records	Number of classes	URL where dataset is available
PlantVillage [47]	An expertly curated dataset by PlantVillage which is a non-profit project by Penn State University and EPFL	Lab-controlled	54,303	38	https://www.tensorflow.org/datasets/catalog/plant_village
PlantDoc	A dataset of internet scraped images which was accepted at ACM India Joint International Conference on Data Science and Management of Data in 2020	Both lab-controlled and real-world	2,598	17	https://github.com/pratikkayal/PlantDoc-Dataset
Rice leaf diseases	A dataset which was created under the supervision of farmers by separating infected leaves of rice into different disease classes	Lab-controlled	120	3	https://archive.ics.uci.edu/ml/datasets/Rice+Leaf+Diseases#
RoCoLe [48]	A dataset of Robusta coffee leaf images which were visually assessed for classification by an expert	Real-world	1,560	5	https://data.mendeley.com/datasets/c5yvn32dzg/2
Cassava leaf disease classification	A dataset of cassava images compiled by farmers and experts at the National Crops Resources Research Institute in collaboration with Makerere University	Real-world	21,367	5	https://www.kaggle.com/c/cassava-leaf-disease-classification/data
Cotton leaf infection	A dataset for cotton leaf disease classification	Both lab-controlled and real-world	1,195	4	https://www.kaggle.com/datasets/raaavan/cottonleafinfection
Dataset of citrus fruit and leaves [49]	A dataset of citrus fruits and leaves images acquired under the supervision of a domain expert	Lab-controlled	759	5	https://data.mendeley.com/datasets/3f83gxmv57/2
Data for: a low shot learning method for tea leaf's disease identification [50]	A dataset of tea leaf's disease images for the paper entitled 'A low shot learning method for tea leaf's disease identification'	Both lab-controlled and real-world	130	3	https://data.mendeley.com/datasets/dbjyfkn6jr/1
Chili plant diseases	A dataset of chili leaf images	Real-world	500	5	https://www.kaggle.com/dhenyd/chili-plant-diseas
Wheat leaf dataset [51]	A dataset of wheat Leaf images collected at at Holeta wheat farm in Ethiopia	Real-world	407	3	https://data.mendeley.com/datasets/wgd66f8n6h/
Banana leaf disease images [52]	A dataset of banana leaf images collected by farmers and verified by plant pathologists	Real-world	1,288	3	https://data.mendeley.com/datasets/rjykr62kdh/1
Guava fruits and leaves dataset [53]	A dataset of guava fruits and leaves collected in Pakistan under the supervision of a domain expert	Real-world	306	4	https://data.mendeley.com/datasets/s8x6jn5cvr/1
PlantaeK [54]	A dataset of leaf images of grapes, cherry, apple, apricot, pear, cranberry, peach, and walnut collected in Jammu and Kashmir	Lab-controlled	2,157	14	https://data.mendeley.com/datasets/t6j2h22jpx/1
Ibean [55]	A dataset of beans leaf images compiled by the Makerere AI laboratory in collaboration with the National Crops Resources Research Institute (NaCRRI) in Uganda	Real-world	1,295	3	https://www.tensorflow.org/datasets/catalog/beans

3.2. Class distribution

In most public datasets for the identification of diseases in plants, we have observed the existence of a class imbalance whereby the number of plant images in some classes is greater than those of other classes. As explained by various researchers, the class imbalance issue in datasets for plant disease detection still prevails because disease lesions in real cultivation fields are less prevalent, and exhausting labor requirements are involved in capturing and annotating leaf images [56], [57]. From Figure 3, it can be observed that common related datasets, including our customized dataset, have different numbers of images in each class. The common datasets represented in Figure 3 not only suffer from the class imbalance issue but also consist of fewer records than our dataset.

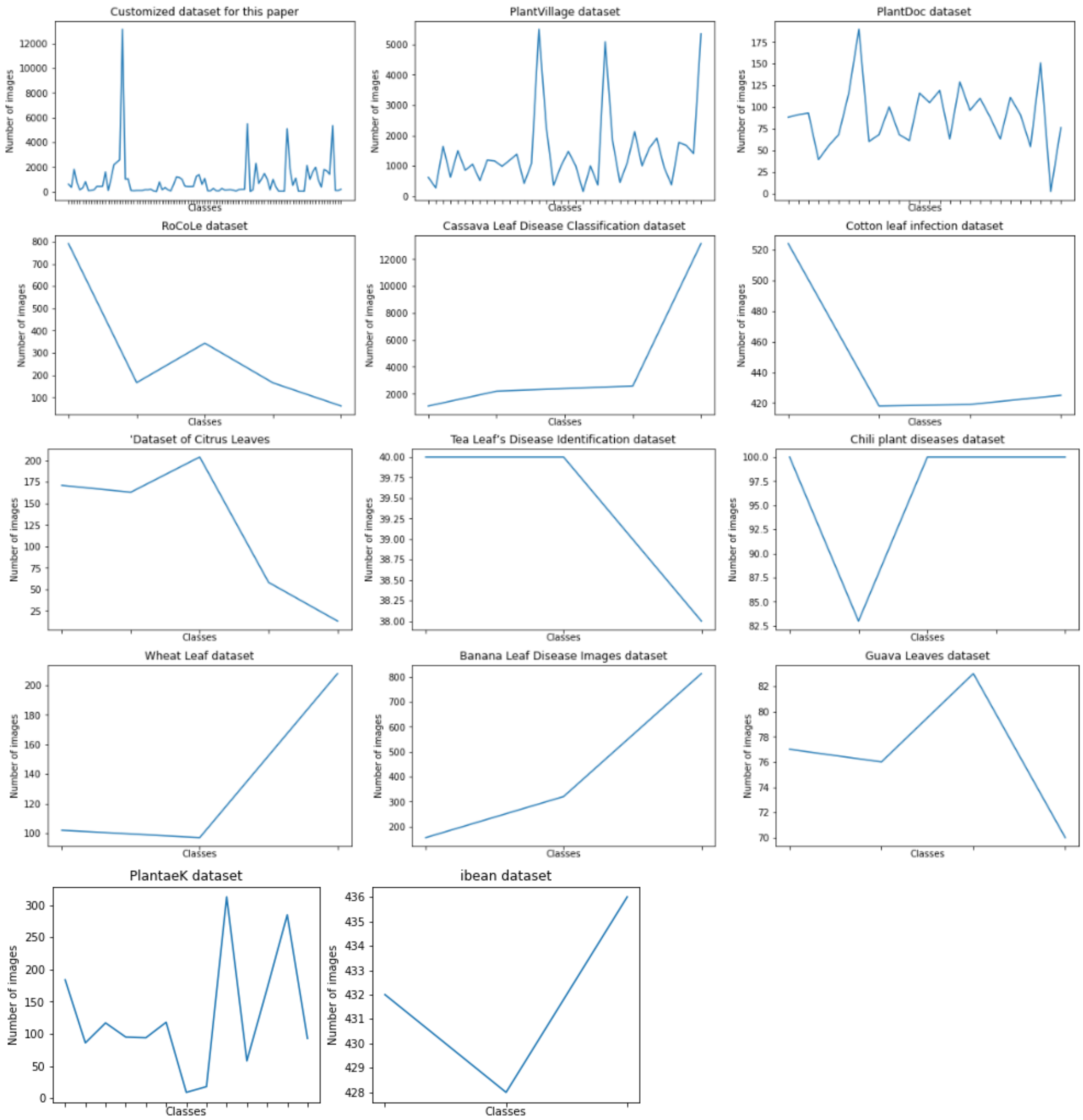

Figure 3. Class imbalance in common plant datasets and the customized dataset

3.3. Data processing

Images of the customized dataset were programmatically selected at random, and split into train and test sets in a ratio of 70% to 30% respectively. The train and test sets comprise 61,259 and 26,311 images respectively, and were loaded from their respective directories using the Keras ImageDataGenerator feature. For the purpose of data augmentation which serves to introduce sample diversity and reduce overfitting, the ImageDataGenerator object was initialized with the parameters shown in Table 3. Additionally, "categorical" class mode was used given that the classification is based on more than 2 classes. The train and test batches were of size 64 each. "steps_per_epoch" was calculated as the number of train images per batch size while "validation_steps" was calculated as the number of test images per batch size.

Table 3. Image augmentation parameters

Parameter	Value
Horizontal flip	True
Vertical flip	True
Width shift range	0.1
Height shift range	0.1
Zoom range	0.2
Rotation range	20

3.4. Transfer learning for classification

In this work, transfer learning utilizing the architectures of four state-of-the-art CNNs models pre-trained on the ImageNet dataset, namely VGG-16, VGG-19, ResNet-50, and ResNet-101 was applied to our customized dataset of leaf images. ImageNet is a large collection of annotated images such as objects, animals and scenes, while transfer learning is the approach that aims to save time and computational resources by reusing features learned from one task in another rather than relearning from scratch. On the other hand, CNN consists of deep, feed-forward artificial neural networks that emulate the way the human vision system works by employing the mechanism of distinguishing one image from another by analyzing input images and then assigning weights to various aspects of each image.

VGG was the runner up of the 2014 ImageNet large scale visual recognition challenge (ILSVRC) while ResNet was the winner of ILSVRC 2015 [58], [59]. The advantage of ResNet over VGG is that it consists of deep networks which do not allow the vanishing gradient problem to occur [59]. Tables 4 and 5 illustrate the VGG and ResNet architectures used in this paper.

Table 4. ResNet-50 and ResNet-101 architectures as per the original paper [59]

Layer name	Output size	50-layer	101-layer
conv1	112×112	7×7, 64, stride 2	
conv2_x	56×56	3×3 max pool, stride 2 [1 × 1,64 3 × 3,64 1 × 1,256] × 3	[1 × 1,64 3 × 3,64 1 × 1,256] × 3
conv3_x	28×28	[1 × 1,28 3 × 3,128 1 × 1,512] × 4	[1 × 1,28 3 × 3,128 1 × 1,512] × 4
conv4_x	14×14	[1 × 1,256 3 × 3,256 1 × 1,1024] × 6	[1 × 1,256 3 × 3,256 1 × 1,1024] × 23
conv5_x	7×7	[1 × 1,512 3 × 3,512 1 × 1,2048] × 3	[1 × 1,512 3 × 3,512 1 × 1,2048] × 3
	1×1	average pool, 100-d FC, softmax	
FLOPs		7.6×10^9	11.3×10^9

Table 5. VGG-16 and VGG-19 architectures as derived from the original paper [58]

ConvNet configuration	
16 weight layers	19 weight layers
input (224×224 RGB image)	
conv3-64	conv3-64
conv3-64	conv3-64
maxpool	
conv3-128	conv3-128
conv3-128	conv3-128
maxpool	
conv3-256	conv3-256
conv3-256	conv3-256
conv3-256	conv3-256
	conv3-256
maxpool	
conv3-512	conv3-512
conv3-512	conv3-512
conv3-512	conv3-512
	conv3-512
maxpool	
conv3-512	conv3-512
conv3-512	conv3-512
conv3-512	conv3-512
	conv3-512
maxpool	
FC-4096	
FC-4096	
FC-1000	
softmax	

The 4 pre-trained models, VGG16, VG19, ResNet50, and ResNet101 were loaded with ImageNet weights, and the final output layer in each base model was removed since it did not correspond to the number of units that our plant disease classification work required. The training images in RGB format were resized to 100×100 and then fed as input to each pre-trained model. Due to the difference between the tasks of ImageNet and ours, we proceeded with fine-tuning the models. All layers of each base model were frozen, and then the following layers of the specific models were set to trainable, provided that that layer was not a batch normalization one:

- The last 4 layers of the VGG-16 model.
- The last 10 layers of the VGG-19 model.

- The last 11 layers of the ResNet-50 model.
- The last 21 layers of the ResNet-101 model.

Table 6 describes the further operations conducted on each model. During the fine-tuning process, the batch normalization layers were kept frozen to prevent the accuracy of the first epoch from decreasing significantly. Dropout layers were also added to the models to randomly set the activation to zero so as to prevent each network from over-learning certain features. Moreover, Adam optimizer was used to enhance performance and speed when training the models. A low learning rate was additionally set to allow the models to learn optimally by not allowing much changes to occur from what was previously learned. Afterwards, to overcome overfitting/underfitting and find a best-fit model, the EarlyStopping callback was used to monitor the validation accuracy of each model such that the training ends if there is no improvement in the performance measure after 10 epochs.

Table 6. Further operations carried out on each pre-trained model

Model	Operations on each model	Further operations on each model
VGG-16 VGG-19	- Add a global average pooling 2D layer	- Define the output layer as a dense layer, with SoftMax activation function, containing 97 neurons - Compile the model using Adam optimizer with a learning rate of 0.0001 and categorical cross-entropy loss
ResNet-50 ResNet-101	- Create a sequential model - Use the pre-trained model as a layer in the sequential model - Add a global average pooling 2D layer - Add a dense layer of 1,024 neurons and ReLU activation - Add a dropout layer of dropout rate of 30%	

4. RESULTS AND DISCUSSION

4.1. Experimental results

To evaluate and compare the performance of the deep transfer learning models for detecting diseases in plants, we plotted the accuracy/loss versus epoch graph for each model as illustrated in Figure 4, and used overall accuracy, precision, f1-score and recall as evaluation metrics, as shown in Table 7, true positive (TP), true negative (TN), false positive (FP) and false negative (FN) respectively.

$$Overall\ Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$Precision = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

$$Recall = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

$$F1 - score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \quad (4)$$

Table 7. Comparing results of models implemented

Performance metric	VGG-16	VGG-19	ResNet-50	ResNet-101
Accuracy	0.8590	0.8591	0.4692	0.4068
Precision	0.8954	0.8966	0.5981	0.7337
Recall	0.8427	0.8377	0.3703	0.1662
F1-score	0.8700	0.8700	0.4564	0.2689

From Table 7, it can be observed that the VGG-16, VGG-19, ResNet-50, and ResNet-101 models are:

- 85.90%, 85.91%, 46.92%, and 40.68% accurate in making a correct prediction respectively.
- able to predict a specific class correctly 89.54%, 89.66%, 59.81%, and 73.37% of the time respectively.
- able to predict 84.27%, 83.77%, 37.03% and 16.62% of the classes correctly out of all time a specific class should have been predicted respectively.

In terms of accuracy, precision, recall and f1-score, VGG-16 and VGG-19 outperform ResNet-50 and ResNet-101. The accuracy and f1-score of both VGG-16 and VGG-19 are more or less similar, as can be deduced from Table 7. F1-score is computed as the weighted mean of precision and recall. Given the existence of the imbalanced class distribution in our customized dataset, f1-score is an ideal metric to evaluate our models as it takes into account both precision and recall. A vast difference between the f1-score of the VGG and ResNet models can be noted from Table 7. The f1-scores of ResNet-50 and ResNet-101 are less than those of the VGG models by 41.36% and 60.11%, respectively. We therefore conclude that VGG-16 and VGG-19 achieve the best performance compared to ResNet-50 and ResNet-101.

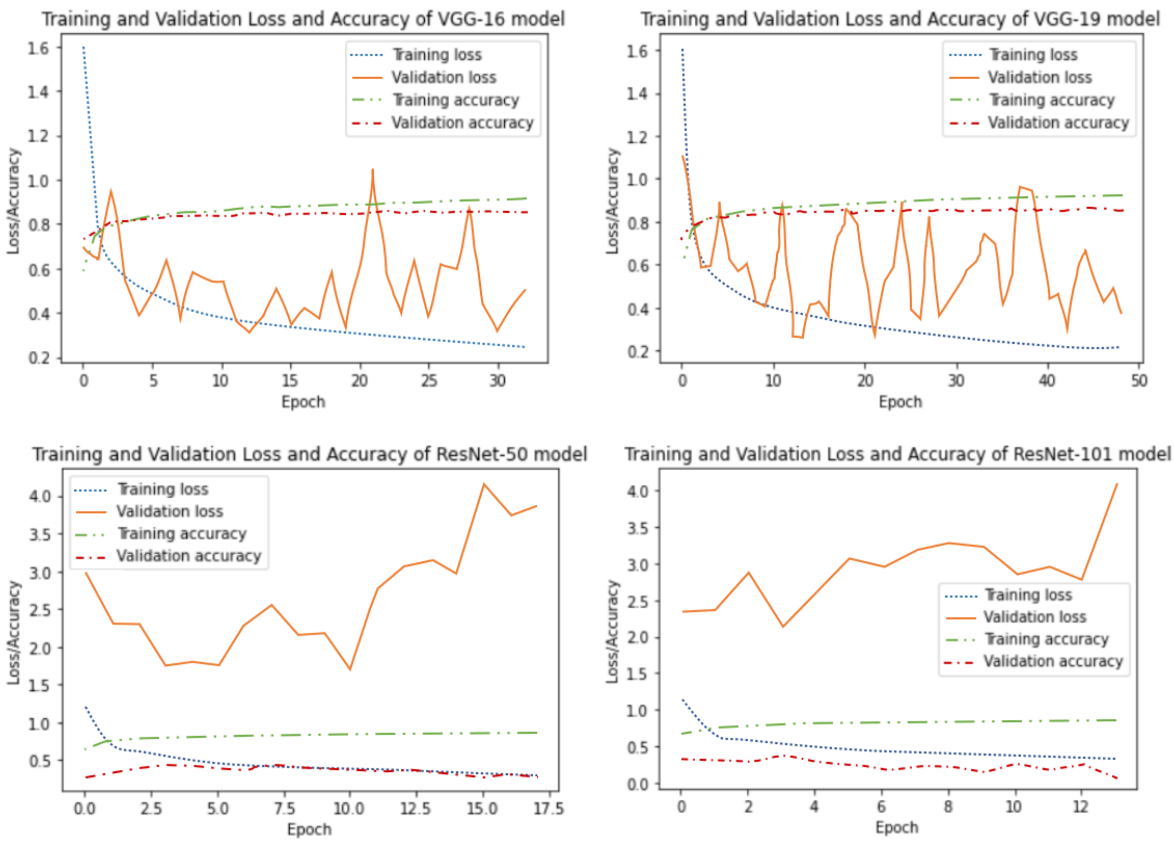

Figure 4. Accuracy/loss versus epoch of each model

4.2. Comparison with existing studies

A comparison of our proposed models with related works is given in Table 8. From the results emanating from Sumalatha *et al*. [60] whereby a subset of the PlantVillage dataset of 11,333 records was used, it can be observed that there is a wide difference between the accuracy of VGG-19 and ResNet-50, which is similar to our case. However, when using the Adam optimizer, as in this paper, that research work obtained a higher accuracy than us for VGG-19 and ResNet-50. Nonetheless, as explained by various researchers [61], those models, as well as other similar state-of-the-art systems with a relatively higher accuracy, are not practical for use in real life because the leaf images studied contain no realistic backgrounds. In another study conducted by Ahmad *et al*. [62] with a dataset of tomato leaf images from real cultivation fields, f1-scores within the range of 77% and 83% were attained for VGG-16 and VGG-19 which is in line with our research. However, that paper utilized fewer records than us, and yielded less accuracy and a lower f1-score for VGG-16 and VGG-19 as compared to our work. Research by Mohameth *et al*. [63] also obtained a higher level of accuracy for VGG-16 than ResNet-50 like us, when classifying diseases in wheat.

Table 8. Comparison between DL studies for plant disease detection

Paper	Dataset	Dataset size	Number of classes	Environment	CNN architecture	Accuracy (%)
Proposed work	Customized dataset	87,570	97	Both lab-controlled and real-world	VGG-16	85.90
					VGG-19	85.91
					ResNet-50	46.92
					ResNet-101	40.68
Sumalatha *et al.* [60]	PlantVillage	11,333	10	Lab-controlled	VGG-19	92.49
					ResNet-50	60.18
Ahmad *et al.* [62]	Tomato leaves	15,216	6	Real world	VGG-16	76.29
					VGG-19	79.60
Mohameth *et al.* [63]	PlantVillage	54,000	36	Lab-controlled	VGG-16	97.82
					ResNet-50	95.38

4.3. Contribution in practice

This work has the potential to make a great contribution towards diagnosing plant diseases in real cultivation fields due to the customized dataset being the widest compilation of plants in both real world and laboratory scenarios. It consists of the largest number of records and the broadest classes of diseased plants to date. We shall publish this dataset, and anticipate that it will be of aid to researchers and people in the agricultural sector.

5. CONCLUSION

This research work has been able to show that, through transfer learning and fine-tuning, VGG-16 and VGG-19 outperform ResNet-50 and ResNet-101 on the customized dataset. The dataset consists of 87,570 leaf images categorized into 97 classes and is able to detect 74 diseases in plants. At the time of working on this paper, no single public dataset with leaf images in both real-world settings and laboratory setups contained more records or classes. As such, no DL models have been able to detect a larger number of plant diseases than this work. Classifying diseases in fields consisting of a wide variety of different plant species is important. In addition, due to the unbalanced number of images in the classes of our dataset, we have analyzed the f1-score along with the accuracy of the models. We concluded that VGG-16 and VGG-19 perform better than ResNet-50 and ResNet-101 as they have a higher f1-score and accuracy. The overall accuracy of VGG-19 and VGG-16 is 85.9%, and the f1-score of VGG-19 and VGG-16 is 87.0%.

6. LIMITATIONS AND FUTURE WORK

As future work, the performance of the models implemented can be enhanced by reducing the class imbalance problem so as to ascertain that there is a comparable number of images in each class. Significantly higher classes can be downsampled while data in relatively small classes of the customized dataset can be augmented through ML techniques such as generative adversarial networks (GAN) with label smooth regularization for curbing the resulting loss function. Additionally, the number of epochs, layers, image size, learning rate and optimizer can be varied.

Furthermore, existing datasets for disease classification in plants still have a limited number of records since collecting and annotating leaf images is difficult. As more data spanning wider varieties of plants and their diseases becomes available over time, more robust and efficient DL models can be developed to better classify diseases in plants. Further analysis will also be required in order to determine whether the models developed can cater for disease detection in individual plant species.

APPENDIX

Table 1. Number of images in each class

Plant	Class	Number of images
Apple	Apple black rot	621
	Apple cedar apple rust	362
	Apple healthy	1,814
	Apple scab	723
Banana	Banana healthy	155
	Banana sigatoka	320
	Banana xanthomonas	814
Basil	Basil wilted	89
	Basil with mildew	109
	Healthy basil	165
Bean	Bean angular leaf spot	432
	Bean healthy	428
	Bean rust	436
Blueberry	Blueberry healthy	1,608
Brassica	Brassica black rot	107
Cassava	Cassava bacterial blight	1,087
	Cassava brown streak disease	2,189
	Cassava green mottle	2,386
	Cassava healthy	2,577
	Cassava mosaic disease	13,158
Cherry	Cherry healthy	1,024
	Cherry powdery mildew	1,052
Chili	Chili healthy	100
	Chili leaf curl	83
	Chili leaf spot	100
	Chili whitefly	100

Table 1. Number of images in each class (continue)

Plant	Class	Number of images
Chili	Chili yellowish	100
	Citrus black spot	171
Citrus	Citrus canker	163
	Citrus greening	204
	Citrus healthy	58
	Citrus melanose	13
Coffee	Coffee healthy	794
	Coffee red spider mite	167
	Coffee rust level 1	344
	Coffee rust level 2	166
	Coffee rust level 3	62
Corn	Corn cercospora leaf spot	579
	Corn common rust	1,202
	Corn healthy	1,162
	Corn northern leaf blight	997
Cotton	Cotton bacterial blight	448
	Cotton curl virus	418
	Cotton fusarium wilt	419
	Cotton healthy	425
Grape	Grape black rot	1,240
	Grape esca black measles	1,383
	Grape healthy	596
	Grape leaf blight isariopsis leaf spot	1,076
Guava	Guava canker	77
	Guava dot	76
	Guava healthy	277
	Guava mummification	83
	Guava rust	70
Coriander	Healthy coriander	272
Kale	Kale with spots	137
Lettuce	Lettuce anthracnose	154
	Lettuce bacterial spot	173
	Lettuce downy mildew	123
	Lettuce soft rot	57
Mint	Mint fusarium wilt	175
	Mint leaf rust	193
	Powdery mildew mint leaf	199
Orange	Orange huanglongbing citrus greening	5,507
Parsley	Parsley leaf blight disease	19
	Parsley leaf spot disease	169
Peach	Peach bacterial spot	2,297
	Peach healthy	675
Pepper	Pepper bell bacterial spot	1047
	Pepper bell healthy	1,482
Potato	Potato early blight	1,000
	Potato healthy	152
	Potato late blight	1,000
Raspberry	Raspberry healthy	413
Rice	Rice bacterial leaf blight	40
	Rice brown spot	40
	Rice leaf smut	40
Soybean	Soybean healthy	5,100
Squash	Squash powdery mildew	1,855
Strawberry	Strawberry healthy	517
	Strawberry leaf scorch	1,109
Tea	Tea leaf blight	40
	Tea red scab	38
Tomato	Tomato bacterial spot	2,127
	Tomato early blight	1,006
	Tomato healthy	1,599
	Tomato late blight	1,982
	Tomato leaf mold	963
	Tomato mosaic virus	379
	Tomato septoria leaf spot	1,801
	Tomato spider mites two spotted spider mite	1,676
	Tomato target spot	1,404
	Tomato yellow leaf curl virus	5,359
Wheat	Wheat healthy	102
	Wheat septoria	97
	Wheat stripe rust	208

REFERENCES

[1] International Plant Protection Convention, "Protecting the world's plant resources from pests: Safe trade, food security and environmental protection with IPPC new adopted standards!," *International Plant Protection Convention*, 2018. https://www.ippc.int/en/news/protecting-the-worlds-plant-resources-from-pests-safe-trade-food-security-and-environmental-protection-with-ippc-new-adopted-standards/ (accessed Aug. 17, 2022).

[2] J. Boulent, S. Foucher, J. Théau, and P. L. St-Charles, "Convolutional neural networks for the automatic identification of plant diseases," *Frontiers in Plant Science*, vol. 10, pp. 1–15, Jul. 2019, doi: 10.3389/fpls.2019.00941.

[3] D. D. Weisenburger, "Human health effects of agrichemical use," *Human Pathology*, vol. 24, no. 6, pp. 571–576, Jun. 1993, doi: 10.1016/0046-8177(93)90234-8.

[4] K. L. Bassil, C. Vakil, M. Sanborn, D. C. Cole, J. S. Kaur, and K. J. Kerr, "Cancer health effects of pesticides: systematic review," *Canadian Family Physician*, vol. 53, no. 10, pp. 1704–1711, 2007.

[5] K.-H. Kim, E. Kabir, and S. A. Jahan, "Exposure to pesticides and the associated human health effects," *Science of The Total Environment*, vol. 575, pp. 525–535, Jan. 2017, doi: 10.1016/j.scitotenv.2016.09.009.

[6] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, pp. 1–10, Sep. 2016, doi: 10.3389/fpls.2016.01419.

[7] Y. Toda and F. Okura, "How convolutional neural networks diagnose plant disease," *Plant Phenomics*, pp. 1–14, Mar. 2019, doi: 10.34133/2019/9237136.

[8] R. W. Risebrough, "Pesticides and bird populations," in *Current Ornithology*, Boston: Springer, 1986, pp. 397–427, doi: 10.1007/978-1-4615-6784-4_9.

[9] H. K. Gill and H. Garg, "Pesticide: environmental impacts and management strategies," in *Pesticides-Toxic Aspects*, InTech, 2014, pp. 187–211, doi: 10.5772/57399.

[10] D. Goulson, "Pesticides linked to bird declines," *Nature*, vol. 511, no. 7509, pp. 295–296, Jul. 2014, doi: 10.1038/nature13642.

[11] F. Sanchez-Bayo and K. Goka, "Pesticide residues and bees–a risk assessment," *PLoS ONE*, vol. 9, no. 4, pp. 1–16, Apr. 2014, doi: 10.1371/journal.pone.0094482.

[12] S. Knillmann and M. Liess, "Pesticide effects on stream ecosystems," in *Atlas of Ecosystem Services*, Cham: Springer International Publishing, 2019, pp. 211–214, doi: 10.1007/978-3-319-96229-0_33.

[13] R. Hu *et al.*, "Long- and short-term health effects of pesticide exposure: A cohort study from China," *PLoS ONE*, vol. 10, no. 6, pp. 1–13, Jun. 2015, doi: 10.1371/journal.pone.0128766.

[14] M. Svensson, R. Urinboyev, A. W. Svensson, P. Lundqvist, M. Littorin, and M. Albin, "Migrant agricultural workers and their socio-economic, occupational and health conditions–a literature review," *SSRN Electronic Journal*, pp. 1–34, 2013, doi: 10.2139/ssrn.2297559.

[15] C. A. Harvey *et al.*, "Extreme vulnerability of smallholder farmers to agricultural risks and climate change in Madagascar," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 369, no. 1639, pp. 1–12, Apr. 2014, doi: 10.1098/rstb.2013.0089.

[16] International Fund for Agricultural Development (IFAD), *Smallholders, food security, and the environment*. Rome: IFAD, 2013.

[17] J. G. A. Barbedo, "Digital image processing techniques for detecting, quantifying and classifying plant diseases," *SpringerPlus*, vol. 2, no. 1, pp. 1–12, Dec. 2013, doi: 10.1186/2193-1801-2-660.

[18] P. H. Flynn, "Plant disease diagnostics. Biotechnology information series," National Central Regional Extension Publication, Iowa State University, Ames, 1994.

[19] F. W. Nutter, "Disease assessment terms and concepts," in *Encyclopedia of Plant Pathology*, New York: John Wiley and Sons, Inc., 2001, pp. 312–323.

[20] A.-K. Mahlein, "Plant disease detection by imaging sensors–parallels and specific demands for precision agriculture and plant phenotyping," *Plant Disease*, vol. 100, no. 2, pp. 241–251, Feb. 2016, doi: 10.1094/PDIS-03-15-0340-FE.

[21] Z. Li *et al.*, "Non-invasive plant disease diagnostics enabled by smartphone-based fingerprinting of leaf volatiles," *Nature Plants*, vol. 5, no. 8, pp. 856–866, Jul. 2019, doi: 10.1038/s41477-019-0476-y.

[22] C. Hillnhütter, A. Schweizer, V. Kühnhold, and R. A. Sikora, "Remote sensing for the detection of soil-borne plant parasitic nematodes and fungal pathogens," in *Precision Crop Protection-the Challenge and Use of Heterogeneity*, Dordrecht: Springer, 2010, pp. 151–165, doi: 10.1007/978-90-481-9277-9_10.

[23] J. S. West, C. Bravo, R. Oberti, D. Moshou, H. Ramon, and H. A. McCartney, "Detection of fungal diseases optically and pathogen inoculum by air sampling," in *Precision Crop Protection-the Challenge and Use of Heterogeneity*, Dordrecht: Springer Netherlands, 2010, pp. 135–149, doi: 10.1007/978-90-481-9277-9_9.

[24] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, Apr. 2017, doi: 10.1080/08839514.2017.1315516.

[25] C. DeChant *et al.*, "Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning," *Phytopathology*, vol. 107, no. 11, pp. 1426–1432, Nov. 2017, doi: 10.1094/PHYTO-11-16-0417-R.

[26] E. Fujita, Y. Kawasaki, H. Uga, S. Kagiwada, and H. Iyatomi, "Basic investigation on a robust and practical plant diagnostic system," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Dec. 2016, pp. 989–992, doi: 10.1109/ICMLA.2016.0178.

[27] Y. Kawasaki, H. Uga, S. Kagiwada, and H. Iyatomi, "Basic study of automated diagnosis of viral plant diseases using convolutional neural networks," in *International Symposium on Visual Computing*, 2015, pp. 638–645, doi: 10.1007/978-3-319-27863-6_59.

[28] M. Brahimi, M. Arsenovic, S. Laraba, S. Sladojevic, K. Boukhalfa, and A. Moussaoui, "Deep learning for plant diseases: detection and saliency map visualisation," in *Human and Machine Learning*, 2018, pp. 93–117, doi: 10.1007/978-3-319-90403-0_6.

[29] L. Bi and G. Hu, "Improving image-based plant disease classification with generative adversarial network under limited training set," *Frontiers in Plant Science*, vol. 11, pp. 1–12, Dec. 2020, doi: 10.3389/fpls.2020.583438.

[30] J. A. Pandian, V. D. Kumar, O. Geman, M. Hnatiuc, M. Arif, and K. Kanchanadevi, "Plant disease detection using deep convolutional neural network," *Applied Sciences*, vol. 12, no. 14, pp. 1–17, Jul. 2022, doi: 10.3390/app12146982.

[31] J. A. Pandian *et al.*, "A five convolutional layer deep convolutional neural network for plant leaf disease detection," *Electronics*, vol. 11, no. 8, pp. 1–15, Apr. 2022, doi: 10.3390/electronics11081266.

[32] A. M. Roy, R. Bose, and J. Bhaduri, "A fast accurate fine-grain object detection model based on YOLOv4 deep neural network," *Neural Computing and Applications*, vol. 34, no. 5, pp. 3895–3921, Mar. 2022, doi: 10.1007/s00521-021-06651-x.

[33] K. Zhang, Q. Wu, A. Liu, and X. Meng, "Can deep learning identify tomato leaf disease?," *Advances in Multimedia*, pp. 1–10, Sep. 2018, doi: 10.1155/2018/6710865.

[34] M. Türkoğlu and D. Hanbay, “Plant disease and pest detection using deep learning-based features,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 27, no. 3, pp. 1636–1651, May 2019, doi: 10.3906/elk-1809-181.
[35] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/j.compag.2018.01.009.
[36] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, “Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks,” *IEEE Access*, vol. 7, pp. 59069–59080, 2019, doi: 10.1109/ACCESS.2019.2914929.
[37] L. G. Nachtigall, R. M. Araujo, and G. R. Nachtigall, “Classification of apple tree disorders using convolutional neural networks,” in *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, Nov. 2016, pp. 472–476, doi: 10.1109/ICTAI.2016.0078.
[38] Y. Jia *et al.*, “Caffe: Convolutional architecture for fast feature embedding,” in *Proceedings of the 22nd ACM international conference on Multimedia*, Nov. 2014, pp. 675–678, doi: 10.1145/2647868.2654889.
[39] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, “Deep neural networks based recognition of plant diseases by leaf image classification,” *Computational Intelligence and Neuroscience*, pp. 1–11, 2016, doi: 10.1155/2016/3289801.
[40] S. Gould, R. Fulton, and D. Koller, “Decomposing a scene into geometric and semantically consistent regions,” in *2009 IEEE 12th International Conference on Computer Vision*, Sep. 2009, pp. 1–8, doi: 10.1109/ICCV.2009.5459211.
[41] A. K. Rangarajan, R. Purushothaman, and A. Ramesh, “Tomato crop disease classification using pre-trained deep learning algorithm,” *Procedia Computer Science*, vol. 133, pp. 1040–1047, 2018, doi: 10.1016/j.procs.2018.07.070.
[42] V. Maeda-Gutiérrez *et al.*, “Comparison of convolutional neural network architectures for classification of tomato plant diseases,” *Applied Sciences (Switzerland)*, vol. 10, no. 4, pp. 1–15, Feb. 2020, doi: 10.3390/app10041245.
[43] J. K. Patil and R. Kumar, “Advances in image processing in image processing for detection of plant diseases,” *Journal of Advanced Bioinformatics Applications and Research*, vol. 2, no. 2, pp. 135–141, 2011.
[44] J. G. A. Barbedo, “Plant disease identification from individual lesions and spots using deep learning,” *Biosystems Engineering*, vol. 180, pp. 96–107, Apr. 2019, doi: 10.1016/j.biosystemseng.2019.02.002.
[45] J. G. A. Barbedo, “Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification,” *Computers and Electronics in Agriculture*, vol. 153, pp. 46–53, Oct. 2018, doi: 10.1016/j.compag.2018.08.013.
[46] J. G. A. Barbedo, “Factors influencing the use of deep learning for plant disease recognition,” *Biosystems Engineering*, vol. 172, pp. 84–91, Aug. 2018, doi: 10.1016/j.biosystemseng.2018.05.013.
[47] D. P. Hughes and M. Salathe, “An open access repository of images on plant health to enable the development of mobile disease diagnostics,” 2015, [Online]. Available: http://arxiv.org/abs/1511.08060.
[48] J. Parraga-Alava, K. Cusme, A. Loor, and E. Santander, “RoCoLe: A robusta coffee leaf images dataset for evaluation of machine learning based methods in plant diseases recognition,” *Data in Brief*, vol. 25, pp. 1–5, Aug. 2019, doi: 10.1016/j.dib.2019.104414.
[49] H. T. Rauf, B. A. Saleem, M. I. U. Lali, M. A. Khan, M. Sharif, and S. A. C. Bukhari, “A citrus fruits and leaves dataset for detection and classification of citrus diseases through machine learning,” *Data in Brief*, vol. 26, pp. 1–7, Oct. 2019, doi: 10.1016/j.dib.2019.104340.
[50] G. Hu, H. Wu, Y. Zhang, and M. Wan, “A low shot learning method for tea leaf’s disease identification,” *Computers and Electronics in Agriculture*, vol. 163, pp. 1–6, Aug. 2019, doi: 10.1016/j.compag.2019.104852.
[51] H. Getachew, “Wheat leaf dataset,” *Mendeley Data*. 2021, doi: 10.17632/wgd66f8n6h.1.
[52] Y. Hailu, “Banana leaf disease images,” *Mendeley Data*. 2021, doi: 10.17632/rjykr62kdh.1.
[53] H. T. Rauf and M. I. U. Lali, “A Guava fruits and leaves dataset for detection and classification of Guava diseases through machine learning,” *Mendeley Data*. 2021, doi: 10.17632/s8x6jn5cvr.1.
[54] V. P. Kour and S. Arora, “Plantaek: A leaf database of native plants of jammu and kashmir,” in *Recent Innovations in Computing*, 2022, pp. 359–368, doi: 10.1007/978-981-16-8248-3_29.
[55] GitHub, “GitHub - AI-Lab-Makerere/ibean: Data repo for the ibean project of the AIR lab,” *GitHub*, 2020. https://github.com/AI-Lab-Makerere/ibean/ (accessed Aug. 18, 2022).
[56] N. M. Nafi and W. H. Hsu, “Addressing class imbalance in image-based plant disease detection: Deep generative vs. sampling-based approaches,” in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, Jul. 2020, pp. 243–248, doi: 10.1109/IWSSIP48289.2020.9145239.
[57] F. P. Boogaard, E. J. van Henten, and G. Kootstra, “Improved point-cloud segmentation for plant phenotyping through class-dependent sampling of training data to battle class imbalance,” *Frontiers in Plant Science*, vol. 13, pp. 1–16, 2022, doi: 10.3389/fpls.2022.838190.
[58] A. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2015, [Online]. Available: http://arxiv.org/abs/1409.1556
[59] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[60] G. Sumalatha, J. R. Singothu, and S. K. Rao, “Transfer learning-based plant disease detection,” *International Journal For Innovative Engineering and Management Research*, vol. 10, no. 3, pp. 469–477, Mar. 2021, doi: 10.48047/IJIEMR/V10/I03/99.
[61] S. Das, S. Pattanayak, and P. R. Behera, “Application of machine learning: a recent advancement in plant diseases detection,” *Journal of Plant Protection Research*, vol. 62, no. 2, pp. 122–135, 2022, doi: 10.24425/jppr.2022.141360.
[62] I. Ahmad, M. Hamid, S. Yousaf, S. T. Shah, and M. O. Ahmad, “Optimizing pretrained convolutional neural networks for tomato leaf disease detection,” *Complexity*, pp. 1–6, Sep. 2020, doi: 10.1155/2020/8812019.
[63] F. Mohameth, C. Bingcai, and K. A. Sada, “Plant disease detection with deep learning and feature extraction using plant village,” *Journal of Computer and Communications*, vol. 8, no. 6, pp. 10–22, 2020, doi: 10.4236/jcc.2020.86002.

BIOGRAPHIES OF AUTHORS

Amatullah Fatwimah Humairaa Mahomodally has a degree in software engineering from the University of Technology, Mauritius. She currently works as an application automation engineer at Accenture, and is passionate about doing research in machine learning. She can be contacted at amahomodally@umail.utm.ac.mu.

Geerish Suddul received his Ph.D. from the University of Technology, Mauritius (UTM). He is currently a Senior Lecturer at the UTM, in the Department of Business Informatics and Software Engineering under the School of Innovative Technologies and Engineering. He has been actively involved in research and teaching since 2005, and currently his research work focuses on different aspects of machine learning such as computer vision and natural language processing. He can be contacted at g.suddul@utm.ac.mu.

Sandhya Armoogum received her Ph.D. from the University of Technology, Mauritius (UTM). She is currently an Associate Professor at the UTM, in the Department of Industrial Systems Engineering under the School of Innovative Technologies and Engineering. She has been actively involved in research and teaching since 2003, and currently her research work focuses on using machine learning in areas such as cyber security, computer vision, agriculture, and data analytics. She can be contacted at sandhya.armoogum@utm.ac.mu.